

MAR 17 2006

PATENT

MS155721.02/MSFTP139USA

CERTIFICATE OF FACSIMILE TRANSMISSION

I hereby certify that this correspondence (along with any paper referred to as being attached or enclosed) is being faxed to 571-273-8300 on the date shown below to Mail Stop Appeal Brief - Patents, Commissioner for Patents, P.O. Box 1450, Alexandria, Virginia 22313-1450.

Date:

3-17-2006


Rebecca Stanford**IN THE UNITED STATES PATENT AND TRADEMARK OFFICE**

In re patent application of:

Applicant(s): Christopher S. Gouge, *et al.*

Examiner: Todd D. Ingberg

Serial No: 09/770,705

Art Unit: 2193

Filing Date: January 26, 2001

Title: SYSTEM AND METHOD FOR CONFIGURING SOFTWARE COMPONENTS

Mail Stop Appeal Brief - Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

RESPONSE TO NOTIFICATION OF NON-COMPLIANT APPEAL BRIEF

Dear Sir:

In response to the Notification of Non-Compliant Appeal Brief, applicants' representative hereby resubmits this corrected brief in connection with an appeal of the above-identified patent application. In the event that any fees may be due, the Commissioner is authorized to charge such fees to Deposit Account No. 50-1063 [MSFTP139USA].

09/770,705

MS155721.02/MSFTP139USA

I. Real Party in Interest (37 C.F.R. §41.37(c)(1)(i))

The real party in interest in the present appeal is Microsoft Corporation, the assignee of the present application.

II. Related Appeals and Interferences (37 C.F.R. §41.37(c)(1)(ii))

Appellants, appellants' legal representative, and/or the assignee of the present application are not aware of any appeals or interferences which may be related to, will directly affect, or be directly affected by or have a bearing on the Board's decision in the pending appeal.

III. Status of Claims (37 C.F.R. §41.37(c)(1)(iii))

Claims 1-24 are currently pending in the subject application and are presently under consideration. Claims 1-24 stand rejected by the Examiner. The rejection of claims 1-24 is being appealed.

IV. Status of Amendments (37 C.F.R. §41.37(c)(1)(iv))

No amendments have been entered subsequent the Final Office Action dated June 28, 2005.

V. Summary of Claimed Subject Matter (37 C.F.R. §41.37(c)(1)(v))**Independent Claim 1**

Independent claim 1 recites a computer implemented system that facilitates software installation comprising: a transformation component that receives one or more configurable data elements, and one or more non-configurable data elements describing the one or more configurable data elements; and a merge component that employs one or more transformation instructions that configures the one or more configurable data elements to facilitate the installation of the one or more configurable data elements into at least one target data set. (See *e.g.*, page 6, line 18-page 8, line 19).

09/770,705

MS155721.02/MSFTP139USA

Independent Claim 9

Independent claim 9 recites a data interpretation system executing on at least one computer, comprising: a data interpretation component that: receives one or more configurable data elements from a configurable module, wherein the one or more data elements include configuration information related to installing the one or more data elements into a software program; applies one or more transformation instructions from the configurable module to the configurable data elements to configure the configurable data elements, and installs the configurable data elements into a target data set based at least in part upon the configuration information. (*See e.g.*, page 7, line 23-page 8, line 19).

Independent Claim 13

Independent claim 13 recites a method that is executed on one or more computer for installing a configurable data set into a target data set, comprising: obtaining one or more data elements from a configurable module; including metadata that describes the configuration options of the configurable data set; obtaining one or more transformation instructions from the configurable module; and applying the one or more transformation instructions to a copy of the one or more data elements from the configurable module to configure the one or more data elements for installation into the target data set. (*See e.g.*, page 10, lines 1-30).

Independent Claim 17

Independent claim 17 recites a computer implemented method for creating a configurable data module, comprising: creating a configurable data set having one or more configurable data elements; creating one or more data structures containing information associated with one or more configurable data elements; and displaying the information to a software program to facilitate installing the configurable data set into the software program. (*See e.g.*, page 11, line 1-page 12, line 18).

09/770,705

MS155721.02/MSFTP139USA

Independent Claim 22

Independent claim 22 recites a system that is executed on one or more computer for installing a configurable data set into a target data set, comprising: a configurable module having configurable data elements representing a configurable data set and non-configurable data elements representing a portion of the configurable data set; a user interface for selecting which configurable data element to modify; and a data interpretation system for receiving the configurable and non-configurable data elements from the configurable module and applying the transformation instructions applicable to the user selections associated with the configurable data elements to enable installing the configurable data set into the target data set. (See e.g., page 6, line 18-page 8, line 19).

Independent Claim 23

Independent claim 23 recites a data packet adapted to be transmitted as a communication signal between at least two computer processes, comprising: a configurable module having: one or more configurable data elements, wherein one or more default values for the one or more configurable data elements are available; one or more non-configurable data elements describing the one or more configurable data elements; and one or more transformation instructions that facilitate configuring the one or more configurable data elements, wherein the instructions are employed to facilitate installation of the one or more configurable data elements into a target data set residing in at least one of the at least two computer processes. (See e.g., page 6, line 18-page 8, line 19 and page 14, line 25-page 15, line 12).

Independent Claim 24

Independent claim 24 recites a computer readable medium having stored thereon a data structure, comprising: a first data field containing one or more configurable data elements, wherein one or more default values for the one or more configurable data elements are available; a second data field containing one or more non-configurable data elements describing the one or more configurable data elements; and a third data field containing one or more transformation instructions that facilitate configuring the one or

09/770,705

MS155721.02/MSFTP139USA

more configurable data elements to load the configurable data elements into a software program. (See e.g., page 6, line 18-page 8, line 19 and page 15, lines 13-25).

VI. Grounds of Rejection to be Reviewed (37 C.F.R. §41.37(c)(1)(vi))

A. Claim 23 stands rejected under 35 U.S.C. §101 as it is alleged that the claimed invention lacks patentable utility.

B. Claims 1-24 stand rejected under 35 U.S.C. §102(b) as being anticipated by Microsoft's Visual C++ version 5.0 as documented in the text book, "Beginning Visual C++5", by Ivor Horton, published March 19, 1997.

VII. Argument (37 C.F.R. §41.37(c)(1)(vii))

A. Rejection of Claim 23 Under 35 U.S.C. §101

Claim 23 stands rejected under 35 U.S.C. §101 because it is alleged that the claimed invention lacks patentable utility. Reversal of this rejection is requested for at least the following reasons. Claim 23 produces a useful, concrete and tangible result, and further the subject claim pertains to the transmission of software code between two or more computer processes.

Because the claimed process applies the Boolean principle [abstract idea] *to produce a useful, concrete, tangible result* ... on its face the claimed process comfortably falls within the scope of §101. *AT&T Corp. v. Excel Communications, Inc.*, 172 F.3d 1352, 1358. (Fed. Cir. 1999) (Emphasis added); See *State Street Bank & Trust Co. v. Signature Fin. Group, Inc.*, 149 F.3d 1368, 1373, 47 USPQ2d 1596, 1601 (Fed.Cir.1998). The inquiry into patentability requires an examination of the contested claims to see if the claimed subject matter, as a whole, is a disembodied mathematical concept representing nothing more than a "law of nature" or an "abstract idea," or if the mathematical concept has been *reduced to some practical application rendering it "useful."* *AT&T* at 1357 citing *In re Alappat*, 33 F.3d 1526, 31 1544, 31 U.S.P.Q.2D (BNA) 1545, 1557 (Fed. Cir. 1994) (emphasis added).

09/770,705

MS155721.02/MSFTP139USA

The Examiner in the Final Office Action (dated June 28, 2005) contends: "[a] communications signal transmitting between two computers a data packet is not tangible." Appellants' legal representative disagrees and submits that the Examiner is misconstruing the requirements necessary to fulfill the conditions for patentability under 35 U.S.C. §101. According to *AT&T Corp. v. Excel Communications, Inc.*, 172 F.3d 1352 (Fed. Cir. 1999), the standard set forth by the Federal Circuit for determining whether claims are directed towards statutory subject matter is whether the claims as a whole can be applied in a practical application to produce a useful, concrete and tangible result. It is the result of the claims as applied in a practical application that is germane to the determination of whether the claims are directed towards statutory subject matter, not whether the underlying means by which the result is effectuated that should be tangible, as the Examiner intimates. It is believed therefore that the subject claim clearly satisfies this legal standard. In particular, independent claim 23 recites: *a data packet transmitted as a communication signal between at least two computer processes, comprising: a configurable module having: one or more configurable data elements, wherein one or more default values for the one or more configurable data elements are available; one or more non-configurable data elements describing the one or more configurable data elements; and one or more transformation instructions that facilitate configuring the one or more configurable data elements, wherein the instructions are employed to facilitate installation of the one or more configurable data elements into a target data set residing in at least one of the at least two computer processes*. Thus, claim 23 elicits a series of independent acts that culminates in a useful, concrete and tangible result – the installation of the one or more configurable data elements into a target data set residing in at least one of the at least two computer processes.

Additionally, the Court of Appeals for the Federal Circuit stated in *Eolas Techs., Inc. v. Microsoft Corp.*, 399 F.3d 1325 (Fed. Cir. 2005):

Title 35, section 101, explains that an invention includes "any new and useful process, machine, manufacture or composition of matter." ... Without question, *software code alone qualifies as an invention eligible for patenting under these categories*, at least as processes. *Id.* at 1338 (emphasis added).

09/770,705

MS155721.02/MSFTP139USA

The subject claim clearly pertains to software code comprising a configuration module, one or more configurable data elements, one or more non-configurable data elements, and one or more transformation instructions encapsulated in a data packet transmitted from one computer process to another computer process, so that the transformation instructions therein can be employed to facilitate installation of the one or more configurable data elements into a target data set residing in at least one of the at least two computer processes. The fact that (i) the data packet that encases the software code during its transmission between two processes, or (ii) the data packet is transmitted as a communication signal between two processes is irrelevant to the fact that it is software code that is contained therein and is being transmitted through utilization of a communication signal. It is submitted that all that is relevant is the fact that software code is being transmitted within the data packet, and that the software code so transmitted produces a useful, concrete and tangible result.

The Examiner in the Advisory Action (dated September 27, 2005) claims that "Examiner merely implement policy not construe policy." *Id.* at page 3. While appellants' legal representative does not entirely disagree with the foregoing statement with regard to construing policy, it is nevertheless asserted that it is the Examiner's duty not to merely implement policy, but rather to implement policy in light of the statutory requirements set forth in Title 35 of the United States Code and the Federal Court's interpretation thereof. This, the Examiner has failed to do.

In addition, the Examiner in the Advisory Action states that "Recent changes in PTO policy has altered the policy on 'carrier waves' in some claim formats." *Id.* While appellants' representative acknowledges that the Patent and Trademark Office is entitled to adopt and implement any policy that it so chooses in satisfying its mandate and to provide guidance on the statute and the Federal Court's interpretation thereof, the policy so adopted must be in accord with the Court's statutory interpretation. Thus, it is appellants' contention that current Patent and Trademark Office policy with respect to carrier waves is at variance with the interpretation of the Federal Courts in this matter.

To date, the Federal Courts, in particular the Supreme Court, in interpreting the ambit of 35 U.S.C. §101 have observed that Congress intended 35 U.S.C. §101 to include "anything under the sun that is made by man." *Diamond v. Chakrabarty*, 447 U.S. 303,

09/770,705

MS155721.02/MSFTP139USA

309 (1980) quoting S.Rep. No. 1979, 82nd Cong., 2nd Sess., 5 (1952). However, despite the broad interpretive scope set forth, the Supreme Court, admittedly, has held that certain categories of subject matter are not entitled to patent protection. In *Diamond v. Diehr*, 450 U.S. 175 (1981) the Supreme Court explained that there are three recognized categories of subject matter for which one may not obtain patent protection, namely “laws of nature, natural phenomena, and abstract ideas.” *Id.* at 185. Subsequent decisions from Federal Circuit have not further limited or expanded these exclusionary categories.

As has been stated *supra*, according to *AT&T Corp. v. Excel Communications, Inc.*, 172 F.3d 1352 (Fed. Cir. 1999), the legal standard set forth by the Federal circuit for determining whether claims are directed towards statutory subject matter is whether the claims can be applied in a practical application to produce a useful, concrete and tangible result. In *AT&T*, the patent at issue described a message record for long-distance telephone calls that included a primary interexchange carrier (“PIC”) indicator, which allowed for differential billing treatment for subscribers. (See *AT&T*, 172 F.3d at 1353). *AT&T*’s claimed process applied Boolean algebra “to determine the value of the PIC indicator, and [applied] that value through switching and recording mechanisms to create a signal useful for billing purposes.” (See *AT&T*, 172 F.3d at 1358). Relying on its holdings in *State Street Bank & Trust Co. v. Signature Fin. Group, Inc.*, 149 F.3d 1368 (Fed. Cir. 1998), *cert. denied*, 525 US 1093 (1999) and *Arrhythmia Research Tech. Inc. v. Corazonix Corp.*, 958 F.2d 1053 (Fed. Cir. 1992), the Court held that the *AT&T* process was patentable subject matter:

In *State Street*, we held that the processing system there was patentable subject matter because the system takes data representing discrete dollar amounts through a series of mathematical calculations to determine a final share price – *a useful, concrete, and tangible result*. See 149 F.3d at 1373, 47 USPQ2d at 1601. In this case, Excel argues, correctly, that the PIC indicator value is derived using a simple mathematical principle (p and q). But that is not determinative because *AT&T* does not claim the Boolean principle as such or attempt to forestall its use in any other application. It is clear from the written description of the ‘184 patent that *AT&T* is only claiming a process that uses the Boolean principle in order to determine the value of the PIC

09/770,705

MS155721.02/MSFTP139USA

indicator. The PIC indicator represents *information* about the call recipient's PIC, *a useful, non-abstract result* that facilitates differential billing of long-distance calls made by an IXC's subscriber. Because the claimed process applies the Boolean principle to produce a *useful, concrete, tangible result* without pre-empting other uses of the mathematical principle, on its face the claimed process comfortably falls within the scope of Section 101. *See Arrhythmia Research Tech. Inc. v. Corazonix Corp.*, 958 F.2d 1053, 1060, 22 USPQ2d 1033, 1039 (Fed. Cir. 1992) ("That the product is numerical is not a criterion of whether the claim is directed to statutory subject matter."). *See AT&T*, 172 F.3d at 1358 (emphasis added)..

In *Arrhythmia*, electrocardiograph signals were input into a computer and filtered and analyzed to determine the average magnitude of the signals. The resulting output signal was then compared to a predetermined level to determine whether the patient was at high risk for a particular arrhythmia. The Court found the claims patentable subject matter stating:

The resultant output is not an abstract number, but is a *signal* related to the patient's heart activity. These claimed steps of "converting", "applying", "determining", and "comparing" are physical process steps that transform one *physical, electrical signal* into another. *The view that "there is nothing necessarily physical about 'signals' " is incorrect. In re Taner*, 681 F.2d 787, 790, 214 USPQ 678, 681 (CCPA 1982) (holding statutory claims to a method of seismic exploration including the mathematically described steps of "summing" and "simulating from"). . . . The computer-performed operations transform a particular input *signal* to a different output *signal*, in accordance with the internal structure of the computer as configured by electronic instructions. "The claimed invention . . . converts one *physical thing* into another *physical thing* just as any other electrical circuitry would do". *Arrhythmia*, 958 F.2d at 1059, 1060 (citations omitted) (emphasis added).

In *State Street*, the Federal Circuit remarked upon its decision in *Arrhythmia* and noted that the transformation of electrocardiographic signals was patentable as "a practical application of an abstract idea . . . because it corresponded to a useful, concrete

09/770,705

MS155721.02/MSFTP139USA

or tangible thing – the condition of a patient's heart.” (*State Street*, 149 F.3d at 1373).

The Federal Circuit also remarked in *State Street* that:

We note that, for the purposes of a Section 101 analysis, it is of little relevance whether [a claim] is directed to a “machine” or a “process,” as long as it falls within at least one of the four enumerated categories of patentable subject matter *State Street*, 149, F.3d at 1373.

As noted above, the Federal Circuit case law supports the contention that carrier waves/signals fall within at least one of the four enumerated categories of patentable subject matter. The Federal Circuit has made clear that signals are physical things, (*see Arrhythmia*, 958 F.2d at 1059, 1060), and as such carrier signals/waves are not naturally occurring phenomena, but rather, manufactured signals which accordingly are patentable products of manufacture in and of themselves. Consequently, in view of the Federal Circuit's holdings with regard to carrier waves/signals it is believed that the Patent and Trademark Office policy that the Examiner is attempting to propound is not in accord with the Federal Courts interpretation of the requirements set forth in Title 35 of the United States Code. Thus, it is submitted that a claim drawn to a data packet transmitted as a communication signal falls squarely within the ambit of statutory subject matter under 35 U.S.C. §101, and thus should be afforded patent protection regardless of a Patent and Trademark Office policy that is demonstrably erroneous and neither reflective of, nor in accord with, the Federal Courts interpretation in this matter. Accordingly, this rejection should be reversed.

B. Rejection of Claims 1-24 Under 35 U.S.C. §102(b)

Claims 1-24 stand rejected under 35 U.S.C. §102(b) as being anticipated by Microsoft's Visual C++ version 5.0 as documented in the text book, “Beginning Visual C++5”, by Ivor Horton, published March 19, 1997 (hereinafter “Horton”). This rejection should be reversed for at least the following reasons. Horton does not disclose or suggest each and every aspect set forth in the subject claims.

09/770,705

MS155721.02/MSFTP139USA

A single prior art reference anticipates a patent claim only if it *expressly or inherently describes each and every limitation set forth in the patent claim*. *Trintec Industries, Inc. v. Top-U.S.A. Corp.*, 295 F.3d 1292, 63 USPQ2d 1597 (Fed. Cir. 2002); *See Verdegaal Bros. v. Union Oil Co. of California*, 814 F.2d 628, 631, 2 USPQ2d 1051, 1053 (Fed. Cir. 1987). The *identical invention must be shown in as complete detail as is contained in the ... claim*. *Richardson v. Suzuki Motor Co.*, 868 F.2d 1226, 9 USPQ2d 1913, 1920 (Fed. Cir. 1989) (emphasis added).

Appellants' claimed invention relates to a system and method for configuring software components that are incorporated into computer programs. In particular, the subject invention relates to creating, describing and configuring software components (that are data sets) that are to be incorporated into software programs (target data sets) such that the components are self-describing in relation to configuration possibilities for the software components. To this end, independent claims 1, 9, 13, 17, 22, 23 and 24 recite similar claim limitations, namely: ... *installation of the one or more configurable data elements into at least one target data set*. Horton does not disclose or suggest these novel features of the invention as claimed.

The Examiner contends that Horton provides substantiation for the rejection of the subject claims. Appellants' legal representative disagrees. Horton discloses the creation and utilization of dynamic link libraries, and specifically that a dynamic link library is a file containing a collection of modules that can be used by any number of different programs. (*See* page 717, lines 1-2). The cited document further discloses that no code from the dynamic link library is ever included in the executable module of any of the programs, such that when one of the programs is executed, the program is loaded into memory, and if the dynamic link library that the program utilizes is not already present in memory, it too *is loaded separately into memory*. (*See* page 718, lines 2-5). Horton also specifically notes that when a program calls a function in a dynamic link library, the operating system will automatically *load the dynamic link library into memory*. Consequently, since the operating system recognizes that the library is already in memory, any program that is subsequently loaded into memory that utilizes the same dynamic link library can use any of the capabilities provided by the same copy of the

09/770,705

MS155721.02/MSFTP139USA

dynamic link library; all that is required is that links between the program and the dynamic link library be established. The cited document further notes that when a dynamic link library is no longer used by any executing program, the operating system will automatically *delete the dynamic link library from memory*. (See page 718, lines 8-15). From the foregoing it is apparent that the dynamic link library as disclosed in Horton is a separate, distinct and transient entity that is loaded into memory to operate in conjunction with a program that is also loaded into memory, and that once the dynamic link library has fulfilled its purpose it is deleted from the memory. The invention as claimed, in contrast, installs and/or loads configurable data elements *into* a target data set through the use of transformational instructions applied to configurable data elements to transform the target data set with the configurable elements obtained from a transformation component, *e.g.*, when the configurable data elements are installed and/or loaded into a target data set there is a transformation in that the target data set is no longer the entity it was prior to utilization of the claimed invention; the target data set now has installed therein configurable data elements.

Additionally, it is evident that a program and the dynamic link library that are loaded into memory always remain distinct entities. In particular, the definitional section provided by the Examiner, *viz.* Term's in the Art, reinforces the perception that the dynamic link library and the program always remain separate and distinct entities in that definition (A) provides that dynamic link libraries are "executable routines [that are] stored separately as files ... [and are] loaded when needed by a program", and definition (B) states that a dynamic link library is "a file containing executable code and data bound to a program at load time, or run time, rather than during linking." The import of these two definitions is that they are an indication that dynamic link libraries are never installed within/into the target/invoking program, but rather that the invoked dynamic link library is merely linked to the invoking program at some point when the functionality of the dynamic link library is required. Thus, it is submitted that the dynamic link library always remains distinct from the target/invoking program, and that the dynamic link library is never installed into the target/invoking program.

In addition, the Examiner in the Response to Arguments section of the Final Office Action dated June 28, 2005, states "the end result of the merge of the content of

09/770,705

MS155721.02/MSFTP139USA

the DLL (software AND/OR data) is the content of the DLL is merged into a target data set (the software component(s)) program which the content is being modified.” *Id.* at page 8. Horton does not teach or suggest that there is a merger of the dynamic link library into the target data set/program, and neither do the definitions upon which the Examiner places reliance. All that Horton (and the definitions provided by the Examiner) discloses is that dynamic link libraries are loaded into memory when needed by an invoking program, and that links are created between the invoking program and the dynamic link library invoked. Nowhere in Horton is there a discussion of merging the dynamic link library into the calling program let alone installing one or more configurable data elements into at least one target data set as recited in the subject claims.

In the Advisory Action (dated September 27, 2005) the Examiner in order to obviate the distinction between the cited document and the invention as claimed contends that the “prior art shows the mechanism [by] which DLLs merge with existing software. The merging is performed with pointers.” *Id.* at page 7. Appellants’ legal representative respectfully disagrees. While it is not disputed that Horton provides links, it is submitted that *linking* a DLL and an invoking program does not connote merger of the two entities. Rather one of ordinary skill in the art would understand linking to imply facilitating a connection or relationship between two entities without subsuming one entity into the other.

The Examiner further states in the Advisory Action that the subject claims are “silent as to the loading of memory when running the invention.” *Id.* Appellants’ representative is well aware that the subject claims do not enunciate loading of memory, but rather installation of one or more configurable data elements into at least one data set. However, appellants’ representative has throughout prosecution merely utilized the passages cited by the Examiner to draw out the distinctions between the cited document and the application as claimed. Now as the Examiner clearly concedes the cited document specifically relates to loading DLLs into memory, whereas the subject claims recite installation of one or more configurable data elements into at least one data set. It is thus submitted that this alone forms a patently distinguishable distinction between the cited document and the invention as claimed.

Additionally, the Examiner in the Advisory Action asserts, “The claimed

09/770,705

MS155721.02/MSFTP139USA

invention makes no mention of the use of memory or garbage collection (often called a sweeper routine).” *Id.* Once again appellants’ representative is cognizant that the invention as claimed does not mention use of memory or garbage collection, but once again it is asserted that appellants’ representative have merely employed those passages that the Examiner has cited to draw out the distinctions between the cited document and appellants’ claimed invention. In particular, it was, and is, appellants’ representative’s intent in highlighting the fact that the operating system automatically loads the dynamic link library into memory, and further automatically deletes the dynamic link library from memory, to further emphasize and clarify the distinctiveness of DLLs throughout the process elucidated in Horton, and specifically the fact that DLLs are never incorporated, subsumed or merged into a target data set as is recited in the subject claims.

Furthermore, it is noted that the standard by which anticipation is to be measured is *strict identity* between the cited document and the invention as claimed, not mere equivalence or similarity. *See, Richardson* at 9 USPQ2d 1913, 1920. This means that in order to establish anticipation under 35 U.S.C. §102, *a single document* must not only expressly or inherently describe *each and every* limitation set forth in the patent claim, but also *the identical invention must be shown in as complete detail as contained in the claim*. Horton alone, despite the supposed “Term’s in the Art” supplied by the Examiner, does not provide the required identity to substantiate this rejection under 35 U.S.C. §102. In particular, the cited document does not provide for the installation of configurable data elements into a target data set thereby rendering a single indivisible entity – a software program that also includes configurable data elements installed therein.

In view of at least the foregoing, it is respectfully requested that the rejection of independent claims 1, 9, 13, 17, 22, 23 and 24 (and claims that depend there from) should be reversed.

09/770,705

MS155721.02/MSFTP139USA

C. Conclusion

For at least the above reasons, the claims currently under consideration are believed to be patentable over the cited references. Accordingly, it is respectfully requested that the rejections of claims 1-24 be reversed.

If any additional fees are due in connection with this document, the Commissioner is authorized to charge those fees to Deposit Account No. 50-1063.

Respectfully submitted,
AMIN & TUROCY, LLP



Himanshu S. Amin
Reg. No. 40,894

AMIN & TUROCY, LLP
24th Floor, National City Center
1900 East 9th Street
Telephone: (216) 696-8730
Facsimile: (216) 696-8731

09/770,705

MS155721.02/MSFTP139USA

VIII. Claims Appendix (37 C.F.R. §41.37(c)(1)(viii))

1. A computer implemented system that facilitates software installation comprising:
a transformation component that receives one or more configurable data elements, and one or more non-configurable data elements describing the one or more configurable data elements; and
a merge component that employs one or more transformation instructions that configures the one or more configurable data elements to facilitate the installation of the one or more configurable data elements into at least one target data set.
2. The system of claim 1, wherein the one or more configurable data elements are stored in a data structure associated with the configurable module.
3. The system of claim 2, wherein the data structure is a metadata item description table.
4. The system of claim 1, wherein the one or more non-configurable data elements are stored in a data structure associated with the configurable module.
5. The system of claim 4, wherein the one or more non-configurable data elements are stored in a metadata item description table.
6. The system of claim 1, wherein the one or more transformation instructions are stored in a data structure associated with the configurable module.
7. The system of claim 6, wherein the one or more transformation instructions are stored in a transformation instruction table.
8. The system of claim 2 wherein the data structure includes at least one of a name of a configurable data element, and a semantic meaning for the configurable data element.

09/770,705

MS155721.02/MSFTP139USA

9. A data interpretation system executing on at least one computer, comprising:
a data interpretation component that:
receives one or more configurable data elements from a configurable module, wherein the one or more data elements include configuration information related to installing the one or more data elements into a software program;
applies one or more transformation instructions from the configurable module to the configurable data elements to configure the configurable data elements,
and
installs the configurable data elements into a target data set based at least in part upon the configuration information.
10. The system of claim 9, further comprising a user interface to enable a user to query the configurable module to determine which of the one or more data elements are configurable.
11. The system of claim 9 further comprising:
a merging component adapted to receive one or more updated configurable data elements from the data interpretation component and adapted to provide the one or more updated configurable data elements to a target data set.
12. The system of claim 9 further comprising:
an authoring schema that describes a configurable module.
13. A method that is executed on one or more computer for installing a configurable data set into a target data set, comprising:
obtaining one or more data elements from a configurable module;
including metadata that describes the configuration options of the configurable data set;
obtaining one or more transformation instructions from the configurable module;
and

09/770,705

MS155721.02/MSFTP139USA

applying the one or more transformation instructions to a copy of the one or more data elements from the configurable module to configure the one or more data elements for installation into the target data set.

14. The method of claim 13, further comprising:
identifying a target data set; and
inserting the updated data elements into the target data set.
15. The method of claim 13 further comprising:
presenting one or more configuration options to a user;
accepting one or more configuration selections from the user; and
selectively configuring the one or more data elements based on the user's configuration selections.
16. A computer readable medium containing computer executable instructions operable to perform the method of claim 13.
17. A computer implemented method for creating a configurable data module, comprising:
creating a configurable data set having one or more configurable data elements;
creating one or more data structures containing information associated with one or more configurable data elements; and
displaying the information to a software program to facilitate installing the configurable data set into the software program.
18. The method of claim 17 wherein creating a configurable data set includes:
identifying one or more attributes of the one or more data elements; and
establishing one or more default values for the attributes of the one or more data elements.

09/770,705

MS155721.02/MSFTP139USA

19. The method of claim 17 wherein creating the one or more data structures further comprises:

identifying one or more locations within a data set that are configurable;

identifying one or more configuration options;

creating one or more instructions concerning how to configure the one or more locations; and

storing the instructions in the one or more data structures.

20. The method of claim 19, wherein the one or more data structures are stored in the configurable data module.

21. A computer readable medium containing computer executable instructions operable to perform the method of claim 17.

22. A system that is executed on one or more computer for installing a configurable data set into a target data set, comprising:

a configurable module having configurable data elements representing a configurable data set and non-configurable data elements representing a portion of the configurable data set;

a user interface for selecting which configurable data element to modify; and

a data interpretation system for receiving the configurable and non-configurable data elements from the configurable module and applying the transformation instructions applicable to the user selections associated with the configurable data elements to enable installing the configurable data set into the target data set.

23. A data packet adapted to be transmitted as a communication signal between at least two computer processes, comprising:

a configurable module having:

one or more configurable data elements, wherein one or more default values for the one or more configurable data elements are available;

09/770,705

MS155721.02/MSFTP139USA

one or more non-configurable data elements describing the one or more configurable data elements; and

one or more transformation instructions that facilitate configuring the one or more configurable data elements, wherein the instructions are employed to facilitate installation of the one or more configurable data elements into a target data set residing in at least one of the at least two computer processes.

24. A computer readable medium having stored thereon a data structure, comprising:
- a first data field containing one or more configurable data elements, wherein one or more default values for the one or more configurable data elements are available;
 - a second data field containing one or more non-configurable data elements describing the one or more configurable data elements; and
 - a third data field containing one or more transformation instructions that facilitate configuring the one or more configurable data elements to load the configurable data elements into a software program.

IX. Evidence Appendix (37 C.F.R. §41.37(c)(1)(ix))

None.

X. Related Proceedings Appendix (37 C.F.R. §41.37(c)(1)(x))

None.